Improvising Label Accuracy for Unsupervised Machine Learning Models

[1]Dr. K.S. Wagh*, [2]Rushikesh Aundhakar*, [3]Smita Muke*, [4]Gopika Nair*, [5]Siddharth Rane*

[1]AISSMS IOIT, SPPU, Pune

Abstract

Converting the unlabeled data into labelled data requires the knowledge of labelling functions, expertise in the domain, high quality input images, training data and various computational algorithms. In order to obtain this, there is a huge demand for domain experts which means lots of human efforts that will eventually incur great costs. Producing large amounts of labelled training data is necessary in order to construct, train, test and deploy an accurate machine learning model. Hence there is an emerging need of correct labelled dataset. There are systems already developed for generating the labelled data from the unlabeled ones. Most of the time the unsupervised model tends to be overconfident i.e. it predicts or assigns wrong labels. In case of small datasets we can manually check the labels whether they are assigned correctly or not but for large datasets we need a proper generalized technique to verify our predicted labels. We propose a technique to increase the labelling accuracy via label smoothing .Hence we are trying to prevent the model from capturing noisy data or from learning incorrect features with the help of soft labels. This will help in reducing the confusion of resemblance of the instances between the different classes. This will thus increase the success rate of the ML algorithms by improving the accuracy of prediction of the probabilistic labels.

Keywords: *Unsupervised Machine Learning, Cross-Entropy, Maximum Likelihood Estimation, Label Smoothing.*

1. Introduction

In these years, there is an immense need of accurate, denoised and labelled data for training the Machine Learning model precisely. In today's time assigning the accurate labels to the unlabeled data is very critical especially when there are no domain experts and also when there is a high requirement of perfectly trained models. In case of the parametric unsupervised learning the probability is assigned on the basis of the set of parameters. The instances are predicted on the basis of mean and standard deviation .Further it uses the Gaussian mixture with the help of Expectation-maximization for predicting class [4]. The main challenge here is to identify whether the instance is assigned with the accurate and correct probabilistic label or not. Assignment of labels is easier in supervised learning as they can verify their labels after the assignment is done using the training model. Practitioners of various unsupervised learning models adapt to weak supervision sources because they provide cheaper sources of labels and also lead to the threat of noisy and heuristic labels [2]. This problem occurs due to the presence of noise in the data . The input which would be unlabeled data may have noise present in it as it comes from various external sources for example the internet. This noise reduces the accuracy of the model causing the model to learn incorrect features. It is possible to detect the incorrectness for a small dataset but in case of a large dataset manually looking at each input and its corresponding output label is practically not possible and feasible. Also at times the model becomes too confident to predict the labels. Regularization is used to prevent the model from being overconfident using regression[5]. To overcome all these issues related to the accuracy of labels we use label smoothing. Label smoothing ensures that the learning model does not become overconfident. It helps to reduce the risk of model being inaccurate using the cross-entropy loss. It makes use of the soft labels instead of the hard labels (0 or 1) to lower the risk of incorrectness. We try to lower the loss of each label thereby lowering the loss of the entire model[2, 4]. Label smoothing works independent of all the assigned datasets, accuracies or architectures. Smoothing is used to make our model robust enough to protect itself from external attacks like Fast Gradient Sign Method (FGSM)[6].

2. Related work

Emerging need of labelled dataset has raised in these years for optimized and accurate results .The aim is to generate efficient and correct labels for all unsupervised machine learning models. This is an active field of research till date. We are mainly focusing on making the labelling of data more accurate by using smoothing technique .Nilaksh Da,Sanya Chaba, Sakshi Gandhi, Duen Horng Chau, Xu Chu[1] proposed a new data programming paradigm called affinity coding for generation of training data automatically. They propose a system called as GOGGLES that would help to label images present in dataset. In this, first the extraction of prototypes is done. Prototypes are nothing but elaborate representative features present in image. Affinity coding finds data instances that are similar and then generates probabilistic class labels which is then benefitted for the purpose of training machine learning models. These labels may or may not be accurate hence there is an need of smoothing technique.

In[2]Alexander Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Re. present Snorkel a distinctive system where users can simply train the model without manually labelling any training data. The only manual operation the user has to do is writing of the labeling functions which will act as a weak supervision source. Here there is possibility of mistakes because it is done manually. Manual intervention can cause inaccuracies. Thus we would try to reduce this issues using smoothing.
In [3],Rafael Müller, Simon Kornblith, Geoffrey Hinton exhibit that label smoothing implicitly adjusts the learned models such that the confidences associated with the predictions are much more aligned with the accuracies of their predictions. They also highlight and make it evident that even if smoothing of the labels increases the accuracy of the teacher network, when teacher networks are trained with label smoothing in case of supervised models, they generate poorer student networks as compared to teacher networks that are trained with hard targets.

In[4], label-smoothing regularization(LSR) technique is proposed that would discourage the model from being too confident. Cross entropy is taken into account for computing the loss. This would better explain the LSR and also establish the desired goal of making the model even more adaptable.
Morgane Goibert, Elvis Dohmatob[6] propose a general framework which is a variation to the general Label Smoothing methods. This is one of the approach that explains the importance of label smoothing . They demonstrate that Label Smoothing improves the robustness of the adversarial networks. Through their experimental setup they prove the result that Label-Smoothing gives a better performance with respect to robustness as compared to natural classifier. In applications where there is a requirement of training neural networks rapidly, this label smoothing technique can be used which will reduce the computation cost. Label Smoothing also increases and enhances the standard accuracy.

Detection of the noisy labels is the preprocessing of smoothing. Jan M. Kohler, Maxmilian Autenrieth, William H. Beluch[7] propose a generalized method in order to detect and re-label the noisy label. They primarily focus on issues arising due to overfitting thereby increasing the performance of the model under consideration. They put forth an iterative process where the uncertainities of the clean and noisy images are represented in the form of distributions. These uncertainties are predicted using various methods such as Deep Ensembles, MC-Dropout and also an amalgamation of both. The uncertainities in the labels are re-labeled by computing the mean and standard deviations. They use Expectation-Maximization algorithm for the purpose of estimating the best distribution for computing uncertainty.

In[8]Zhilu Zhang, Mert R. Sabuncu, put forth a loss function that generalizes the Mean Absolute Error and Categorical Cross entropy. This proposed loss function is highly efficient and capable such that it can be applied with any Deep Neural Network algortihms so that there is good performance in scenarios involving noisy labels. They verify noise robustness on several datasets that have both open-set and closed-set noise scenarios. This is an process carried out as an integral part of label smoothing.

3. Proposed Methodology

In various unsupervised learning models due to the absence of accurate labelled dataset the predicted labels may vary from the expected labels. As there is no source of identifying the precise assigned label to the instance it is difficult to verify the results. Sometimes these unsupervised models may become overconfident and predict the wrong labels. Hence we propose the technique of smoothing for such models. This technique would help augment the reliability of prediction models.

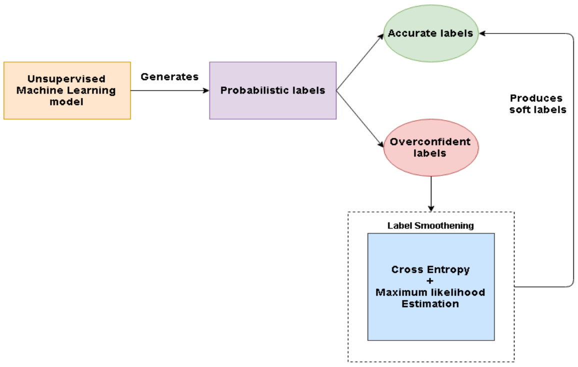

Fig. 1: Working of Label Smoothing

In the domain of machine learning when determining the error rates that are present in the range R[0,1] the terms cross entropy or log loss resolve to the identical context. We are proposing the usage of smoothing using cross-entropy loss. It is also termed as Log-Loss which helps in measuring the performance of classification model. Classification is as follows:

1. **Binary Classification (Binary Labeling):**
Classification is based between two classes for the input instance. The score is predicted and then the input instance is assigned to either of the two categories or classes. Higher computed score indicates positive class and the lower score indicates negative class. We have predefined threshold values to indicate the resemblance of the input instance to one of the class. This classification

thereby assigns the input to either positive or negative class[11].

2. Multi-Class Classification(Multi-class Labeling):

The classification is based on more than two classes. The score is computed and then compared with the thresholds of all the classes, the input having the maximum likelihood threshold value is assigned to that particular class .

This assignment of input instances to classes is nothing but assigning labels to these instances. These labels assigned may become overconfident at times predicting an accuracy as 0 or 1. Such models can be inaccurate in case of unsupervised model as there is no means of verifying theses assigned labels. Therefore we suggest the use of cross-entropy and maximum likelihood estimation in order to reduce the inaccuracies.

A. Cross Entropy

We use the concept of entropy as we are dealing with uncertainties while predicting probability of instance belonging to a particular class. Entropy is a term that derives information from a random variable consisting of various features. It also provides a way to calculate the average amount of data required to predict the event from a particular probabilistic distribution. Entropy is being denoted as H() , where this function is responsible for computing information for a particular random variable Y and probabilistic distribution P[12] of that variable.

$$H(Y) = -\sum y \text{ in } Y\; P(y) * \log(P(y))$$

The entropy can be low or high depending on the probability distribution .When there is low entropy it denotes that the probabilistic distribution was skewed and whereas high entropy denotes that the probabilistic distribution was balanced.

Taking into account the concept of entropy and information obtained we further use cross-entropy in order to calculate the difference between the two different probability distributions. Suppose P and Q are the two probabilistic distributions over which the cross-entropy is to be calculated. Cross-entropy is denoted as H(P,Q) where P may be the target distribution and Q is the approximation of P.

$$H(P,Q) = -\sum y\ in\ YP(y) * loglog(Q(y))$$

Where P(y) is the probability of the event y occurring in P, Q(y) it also denotes the probability of event y in Q explaining that our obtained results are in bits. Negative sign guarantees us that the result would always be either positive or zero. We consider the distributions P and Q for computations.

Mathematical representation of cross-entropy for discrete values

$$H(a,b) = -\sum_{y \in Y} a(y) \log b(y)$$

B. Maximum Likelihood Estimation

Maximum Likelihood Estimation is the another important process for determining the values for the existing parameters of model that perfectly describes the model. In case of Gaussian distribution it has two parameters mean and standard deviation. Here we take into consideration the parameters i.e. the features to predict the maximum likelihood. The features leading to the maximum value of feature set giving the max of the mean and deviation is selected as the best suited result. It mainly focuses on finding the best values of the mean and the standard deviation that lead to formation of best fit curve[13].

C. Cross Entropy + Maximum Likelihood Estimation

For calculating the actual loss and computing the correct accuracy we have :

1. a as the distribution for the ground truth label for feature x.
2. bθ as the distribution for the predicted label for feature x.

The ground truth distribution a(y|xi) is given by :

$$a(y|x_i) = \begin{cases} 1 \; if \; y = \; y_i \\ 0 \; otherwise \end{cases}$$

The entropy is calculated for all the features x of a every input instance in the given probabilistic distributions. This can also be termed as log loss.

$$H_i(a, b_\theta) = -\sum_{y \epsilon Y} a(y|x_i) log b_\theta(y|x_i)$$
$$= -log b_\theta(y_i|x_i)$$

We now compute the sum of all the log loss :

$$L = \sum_{n}^{i=1} log b_\theta(y_i|x_i)$$

Thereby we compute the optimization goal: In this step we finally compute the accurate labels with appropriate accuracy. In other words we can say that at this stage we obtain the soft labels for all the instances that previously were hard labels.

$$L = argmin\theta \sum H_i(a, b_\theta)$$

$$L = argmin\theta - \sum \log q_\theta \, (y_i|x_i)$$

Algorithm :

```
Smoothing(a,b)
{
        a = ground truth label
        b = predicted label for a
        x = feature
        $\theta$  = parameter
        cross entropy $= -\sum p(x) * \log b(x)$
```

$$L = -\sum \log b_{\theta}(x)$$

$$argmin\, L = argmin - \sum_{\theta=1}^{n} L$$

}

This aids in reducing the divergence of the predicted probability from the actual labels. Thus cross entropy and maximum likelihood estimation would solve the model's overfitting problem.

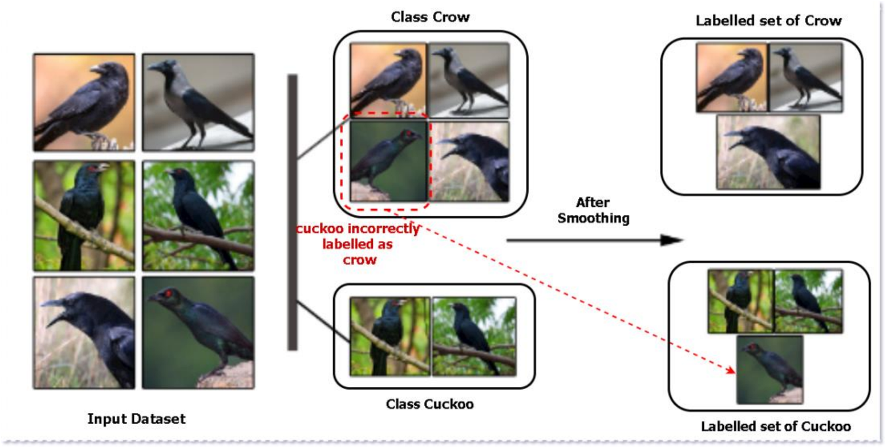

Fig. 2: Label Smoothing using cross entropy and maximum likelihood estimation

For Example: Prediction for a particular input instance may be done by the model as 0.9 whereas its actual value is 0.6. If we do not implement the usage of label smoothing in our model then our output vector label will be computed as: output[1,0,0] . These vector values seem to be quite overfit or overconfident, which is not good for an model. To reduce this unwanted overfitting values we use label smoothing and cut down these output values as output[0.933,0033,0.033] which are more accurate values.

4. Expected Outcome

Labels assigned to the unlabeled instances of an unsupervised model will be accurate. The loss calculated will also be useful to prevent the model from being overconfident which means to avoid the over-fitting of the data. This is nothing but a way of smoothing the labels by converting the hard labels to soft labels. We expect that the combination of cross-entropy and maximum likelihood estimation would smoothen the inaccurate labels. This would also enhance the learning speed of the multi-class neural networks. We expect that this proposed method would reduce the uncertainties and this generalized method would improve labelling technique of unsupervised models. This will be beneficial for the domain experts, ML algorithms and organizations.

5. Conclusion

The problem of having an accurate, efficient and labelled dataset has become very serious for different unsupervised machine learning algorithms. Domain experts also require a precise labelled dataset. In this paper we propose that in order to improve the efficiency and the accuracy of the parametric unsupervised machine learning model, label smoothing paradigm can be implemented. Here we suggest the use of cross entropy along with the maximum likelihood estimation for label smoothing. Cross entropy computes the loss. This computed loss is helpful to improve the accuracy of the model by converting the hard labels to soft labels. The predicted labels would thereby be even more correct and this makes the model more acceptable.

Acknowledgement

The authors would like to thank Prof. Kishor Wagh, AISSMS IOIT , Pune University for the guidance, informative discussions and suggestions for all the methodologies reported in the paper.

References

[1]. Nilaksh Das, Sanya Chaba, Sakshi Gandhi, Deun Horng Chau, Xu Chu, “GOGGLES: Automatic Training Data Generation with Affinity Coding’, Cornell University, March 11, 2019.

[2]. A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, and C. Re, “Snorkel: Rapid training data creation with weak supervision”, Proceedings of the VLDB Endowment.

[3]. Rafael Müller, Simon Kornblith, Geoffrey Hinton. When does label smoothing help ? published on 5 Dec , 2019 in 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

[4]. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, Zbigniew Wojna, “Rethinking the Inception Architecture for Computer Vision”.

[5]. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, “An Introduction to Statistical Learning(ISL)”.

[6]. Morgane Goibert, Elvis Dohmatob Adversarial Robustness via Label-Smoothing.

[7]. Jan M. Kohler,Maximilian Autenrieth,William H. Beluch proposed “Uncertainty Based Detection and Relabeling of Noisy Image Labels” on IEEE Xplore.

[8]. Zhillu Zhang, Mert R.Sabuncu “Generalized cross entropy loss for training deep neural networks with noisy labels”.

[9]. https://towardsdatascience.com/label-smoothing-making-model-robust-to-incorrect-labels-2fae037ffbd0/

[10]. https://leimao.github.io/blog/Cross-Entropy-KL-Divergence-MLE/

[11]. “Amazon Machine Learning: Developer Guide”, Amazon Web Services, Inc. and/or its affiliates, **(2020)**.

[12]. David J.C. MacKay, “Information Theory, Inference, and Learning Algorithms”, Cambridge University Press **(2003)**.

[13]. Myung, In Jae. “Tutorial on maximum likelihood estimation.” Journal of mathematical Psychology 47.1 **(2003)**: 90-100.